A

Project Report on

# **Library Management System**

**(Core Module: 4)**

Submitted in partial fulfillment of completion of the course

Advanced Diploma in IT, Networking and Cloud Computing

Submitted by:

**Archana Kumari**

**Susma Kumari Singh**

**Aashika Raj**

Under Guidance of:

**Arpita Roy (Edunet)**

|  |  |  |
| --- | --- | --- |
| IBM-Logo - Chicago Innovation | DGT introduces high end diploma courses - digitalLEARNING Magazine | Edunet Foundation-Delhi- CSR Organization profile |

Year 2023

**PROJECT OVERVIEW**

Abstract

Acknowledgement

Team Composition and Workload Division

Table of Contents

1. Introduction to Problem
2. Proposed Solution
3. Requirements

4.1 Technology Stack

4.2 Hardware

4.3 Software

4.4 Deployment Environment

5. User Requirements

6. Design Documentation

7. Implementation Details

8. Testing

9. Deployment

10. Future Scope

11. Conclusion

Appendix A Screenshot of Project

References

**ABSTRACT**

The library Management System is a web-based application built using the Django web framework, designed to streamline and automate various aspects of library management. This system facilitates efficient HR and administrative operations by providing a user-friendly interface for managing student data, attendance, leave requests, and performance evaluations.

1. Employee Information Management:

- The system allows HR personnel to easily add, update, and view admin and student details such as personal information, contact details, and job-related data.

- Role-based access control ensures that only authorized users can modify sensitive student information.

2. Reporting and Analytics:

- Built-in reporting tools provide insights into various aspects of library management, helping organizations make data-driven decisions.

3. Security and Data Protection:

- The system ensures data security by implementing user authentication, authorization, and encryption of sensitive information.

- Regular backups and data recovery mechanisms are in place to prevent data loss.

4. User-Friendly Interface:

- The system features an intuitive, user-friendly interface that makes it easy for both student and administrators to navigate and use.

The library Management System in Django offers a comprehensive solution for organizations looking to streamline their HR and administrative processes. By centralizing student data and automating routine tasks, it enhances efficiency, reduces paperwork, and improves overall workforce management.

**ACKNOWLEDGEMENT**

We would like to express our sincere gratitude to all those who have supported and contributed to the successful completion of this project. Your assistance and encouragement have been invaluable throughout this journey.

First and foremost, we want to thank my project supervisors, Arpita Roy(Edunet) and Sayanti Manna(Edunet), for their guidance, expertise, and unwavering support. Their insights and feedback have been instrumental in shaping the direction of this project.

We would also like to extend my appreciation to my fellow classmates who provided valuable input, shared resources, and engaged in stimulating discussions that enriched the project. Your collaborative spirit was a driving force behind our achievements.

Furthermore, we want to acknowledge our friends and family for their patience, understanding, and encouragement throughout this endeavour. Your support provided the motivation we needed to see this project through to its completion.

Last but not least, we are grateful to the entire faculty and staff of NSTI Kolkata for providing a conducive learning environment and the necessary resources to undertake this project.

This project has been a rewarding learning experience, and we are thankful for the collective efforts of everyone involved. Your support has been instrumental in making this project a reality.

Thank you all for being a part of this journey.

Archana Kumari, Susma Kumari Singh, Aashika Raj

NSTI KOLKATA

03/10/2023

**TEAM COMPOSITION AND WORKLOAD DIVISION**

The team composition for the "Library Management System" project can vary depending on the project's complexity, goals, and scope. This is the team composition:

**TEAM COMPOSITION**

1. Aashika Raj (Leader):

* Responsible for overseeing the project's planning, execution, and delivery, coordinate tasks, manage timelines, and ensure that the project stays on track and documentation.

2. Susma Kumari Singh (Developers):

* List the names and contact information of all developers on the team.
* Specify their roles and responsibilities within the project.

3. Archana Kumari (Testers/QA Team):

* Include names and contact information of individuals responsible for testing and quality assurance.
* Explain their roles in ensuring the project meets quality standards.

**WORKLOAD DIVISION**

Detail how the workload is divided among the team members, highlighting their responsibilities and tasks:

1. Susma Kumari Singh:- Project Planning and Conceptualization
2. Aashika Raj:- web developer
3. Archana Kumari Singh :- Design,Testing and quality assurance

All team members participated in designing various scenarios, conducting simulations, and evaluating system responsiveness.

Documentation and Report Writing:

Lead the documentation process, with contributions from all team members for their respective areas of expertise.

**INTRODUCTION TO PROBLEM**

There are many challenges, such as manual bookkeeping, time-consuming cataloging, difficulties in tracking book availability, and managing memberships. Additionally, the rising demand for digital resources and online services necessitates a robust and integrated system that can handle both physical and digital collections efficiently.

* LMS allows librarians to create and maintain a digital catalog of all library resources, making it easier for users to search and locate materials
* Online Public Access Catalog (OPAC): Users can search the library's collection, check availability, place holds, and even renew items online through OPAC interfaces.
* Digital Resource Management: LMS can manage electronic resources such as e-books, journals, and multimedia materials, enabling seamless integration of digital and physical collections.
* Member Management: LMS maintains detailed records of library members, including their borrowing history, preferences, and contact information

.

* Reports and Analytics: Library staff can generate various reports and analyze data to make informed decisions, such as collection development and resource allocation.
* Circulation Management: LMS automates the borrowing and returning of books, tracks due dates, and sends overdue notifications, ensuring smooth circulation processes.

**PROPOSED SOLUTION**

**Benefits of Library Management Systems:**

Implementing an LMS offers several advantage,including:

* Lack of communication: To solve this problem, it is important to establish clear communication channels between the project team, the stakeholders, and the employees. This can be done through regular meetings, emails, and other forms of communication. It is also important to be open and honest about the project's progress and challenges.
* Unrealistic expectations: To avoid this problem, it is important to set realistic expectations for the project at the outset. This includes setting realistic timelines, budgets, and goals. It is also important to be flexible and willing to adjust the project plan as needed.
* Efficiency: Streamlines processes, reducing manual effort and saving time.
* Accuracy: Minimizes errors in cataloging, circulation, and member management.
* User Experience: Enhances the user experience by providing easy access to resources and online services.
* Data Security: Ensures the security and confidentiality of library data.
* Resource Utilization: Facilitates better utilization of library resources and budget allocation.

**REQUIREMENTS**

**TECHNOLOGY STACK**

* Backend Framework: Django is the primary backend framework for this project due to its robust features for web application development.
* Frontend Framework: You can use Django's templating engine for rendering dynamic web pages.
* Programming Languages: Python, HTML, CSS, and JavaScript

**HARDWARE**

* Server: A dedicated server or cloud-based virtual machine (VM) to host the Django application.
* CPU and RAM
* Storage: Adequate storage space for the database, application code, and uploaded files.
* Network Infrastructure: Reliable internet connectivity, firewalls, and security measures to protect the system.

**SOFTWARE**

* Operating System
* Django: Install the Django framework, along with any necessary packages and dependencies. Use virtual environments to manage Python dependencies.
* Development Tools: Visual Studio Code
* Version Control Tools: GitHub or GitLab
* Web Server

**DEPLOYMENT ENVIRONMENT**

* The size and complexity of the project
* The security requirements of the project
* The budget for the project
* The technical expertise of the team

**USER REQUIREMENTS**

* Mobile Accessibility or PCs
* Internet Access
* Supported Browser
* Agreement to Terms

**DESIGN DOCUMENTATION**

****

**IMPLEMENTATION DETAILS**

1. Implementing an Library management system project in Django involves a series of steps and tasks to create, develop, and deploy the system. Below are the implementation details and steps involved in building such a project:
2. Create a Django project. This can be done by running the following command in a terminal:

django-admin startproject librarymanagement

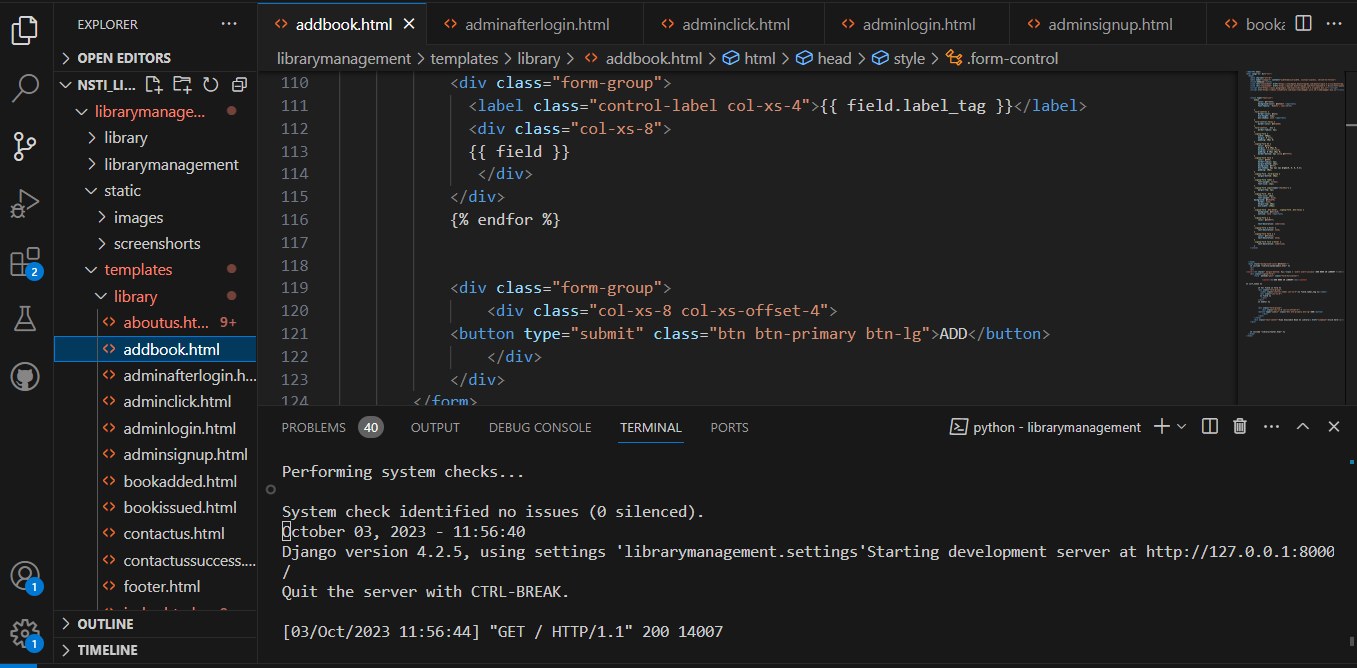
1. Create an app for the employee management system. This can be done by running the following command:

python manage.py startapp library

1. Define the models for the Library management system. This can be done by creating a file called models.py in the library app. The models define the data that will be stored in the database.
2. Create the views for the employee management system. This can be done by creating a file called views.py in the library app. The views define the functionality of the web application.
3. Create the templates for the library management system. This can be done by creating a folder called templates in the library . The templates define the look and feel of the web application.
4. Configure the database. This can be done by editing the settings.py file in the library project. The settings.py file defines the configuration settings for the Django project.
5. Migrate the database. This can be done by running the following command:
   1. python manage.py migrate
6. Run the development server. This can be done by running the following command:
   1. python manage.py run server
7. User Authentication:
   1. Implement user authentication and authorization using Django's built-in authentication system.
   2. Define user roles (e.g., admin, student) and permissions based on project requirements.
8. Create Views and Templates:
   1. Develop views and templates for different sections of the employee management system, such as admin and student profiles, leave requests, tasks, and document management.
   2. Ensure that templates are user-friendly and responsive.
9. Design the User Interface:
   1. Style the user interface using CSS and possibly a frontend framework like Bootstrap, if desired.
   2. Ensure a consistent and visually appealing design across all pages.
10. Implement Functionality:
    1. Implement CRUD (Create, Read, Update, Delete) functionality for student profiles, departments, job titles, leave requests, tasks, and other relevant data.
    2. Develop features for leave and attendance management, including leave request submission and approval workflows.
    3. Create forms for adding and editing employee information.
11. Testing:
    1. Implement unit tests and integration tests to ensure the system's functionality is working as expected.
12. Security and Privacy:
    1. Implement security measures to protect sensitive data, including secure user authentication, role-based access control, and data encryption.

**TESTING**

**This is my overall how to create project image**

****

**This is index page code input**

<!DOCTYPE html>

{% load static %}

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<title>Library Management System</title>

<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css">

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"></script>

<style type="text/css">

    .bs-example{

        margin: 0px;

    }

.navbar-brand{

  font-size: 20px;

  font-family: sans-serif;

}

.col-lg-6{

  padding-left: 20%;

}

.jumbotron{

  margin-bottom: 0px;

  align-items: center;

      width: 100%;

      height: 100%;

      margin-top: 0px;

      margin-bottom: -100px;

      background: linear-gradient(to bottom, rgba(89, 195, 233, 0.5) 0%, rgba(31, 37, 38, 0.5) 100%),url("{% static "images/aboutbackgroundimage1.jpg" %}");

      background-size: cover;

      background-position: center;

      background-repeat: no-repeat;

      text-align: center;

      font: 1em sans-serif;

      color: black;

      font-size: large;

}

.my-5{

  background-color: rgb(165, 194, 234);

}

.d {

  text-align: center;

  align-items: center;

      width: 100%;

      height: 100%;

      margin-top: 0px;

      margin-bottom: -100px;

      background: linear-gradient(to bottom, rgba(89, 195, 233, 0.5) 0%, rgba(31, 37, 38, 0.5) 100%),url("{% static "images/aboutbackgroundimage.jpg" %}");

      background-size: cover;

      background-position: center;

      background-repeat: no-repeat;

      text-align: center;

      font: 1em sans-serif;

      color: black;

      font-size: large;

}

.d img:hover{

  border-color: rgb(178, 216, 9);

}

</style>

</head>

<body>

{% include "library/navbar.html" %}

<div id="carouselExampleIndicators" class="carousel slide" data-ride="carousel">

  <ol class="carousel-indicators">

    <li data-target="#carouselExampleIndicators" data-slide-to="0" class="active"></li>

    <li data-target="#carouselExampleIndicators" data-slide-to="1"></li>

    <li data-target="#carouselExampleIndicators" data-slide-to="2"></li>

  </ol>

  <div class="carousel-inner">

    <div class="carousel-item active">

      <img class="d-block w-100 " height="500px" src="{% static "images/3.jpg" %}" alt="First slide">

    </div>

    <div class="carousel-item">

      <img class="d-block w-100 " height="500px" src="{% static "images/2.jpg" %}" alt="Second slide">

    </div>

    <div class="carousel-item">

      <img class="d-block w-100 " height="500px" src="{% static "images/4.jpg" %}" alt="Third slide">

    </div>

  </div>

  <a class="carousel-control-prev" href="#carouselExampleIndicators" role="button" data-slide="prev">

    <span class="carousel-control-prev-icon" aria-hidden="true"></span>

    <span class="sr-only">Previous</span>

  </a>

  <a class="carousel-control-next" href="#carouselExampleIndicators" role="button" data-slide="next">

    <span class="carousel-control-next-icon" aria-hidden="true"></span>

    <span class="sr-only">Next</span>

  </a>

</div>

<div class="jumbotron">

  <h1 style="text-align:center" class="display-4">Hello, Student!</h1>

  <p class="lead">A service dedicated to Admin and Student. We provide online resources, professional support

    and guidance to all our students whenever, and from wherever they have chosen to study.</p>

    <div class="container mt-3">

      <center><div class="card" style="width:400px">

     <div class="card-body">

       <p class="card-text">

       <img src="{% static "images/admin1.avif" %}" width="90%" height="50%" alt=""><br><br>

        <a class="btn btn-primary btn-lg" style="text-align:center" href="/adminclick" role="button">Admin</a>

         </div>

       </div>

       <br>

      <div class="container mt-3">

       <div class="card" style="width:400px">

      <div class="card-body">

        <p class="card-text">

              <img src="{% static "images/student1.jpg" %}" width="90%" height="50%" alt=""><br><br>

              <a class="btn btn-primary btn-lg" href="/studentclick" role="button">Student</a></center>

          </div>

        </div>

        <br><br>  <br><br>

        <div class="d">

          <h1>Recent works</h1> <br>

          <a href="#"><img src="{% static "images/pick1.jpg" %}" width="300px" height="200px" border="4"></a>

          <img src="{% static "images/pick2.jpg" %}" width="300px" height="200px" border="4">

          <img src="{% static "images/pick3.jpg" %}" width="300px" height="200px" border="4">

          <img src="{% static "images/pick4.jpg" %}" width="300px" height="200px" border="4">

          <img src="{% static "images/pick5.jpg" %}" width="300px" height="200px" border="4">

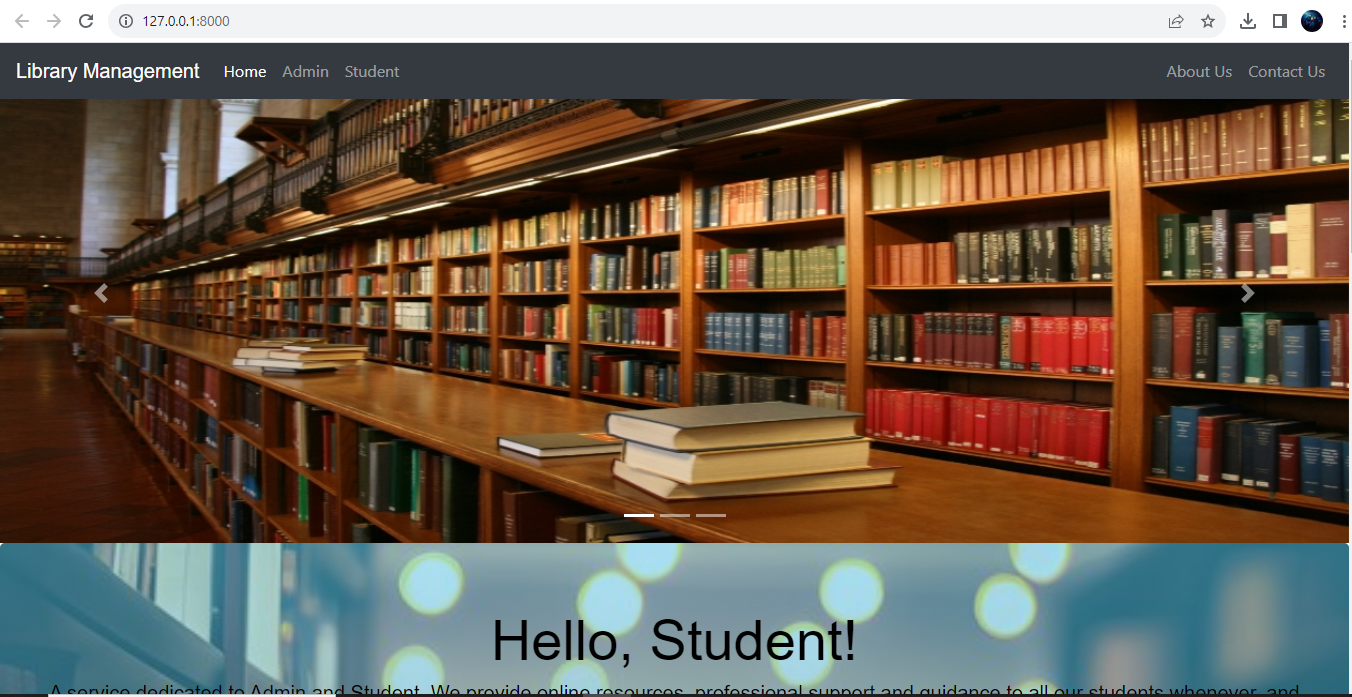
          <img src="{% static "images/pick6.jpg" %}" width="300px" height="200px" border="4">

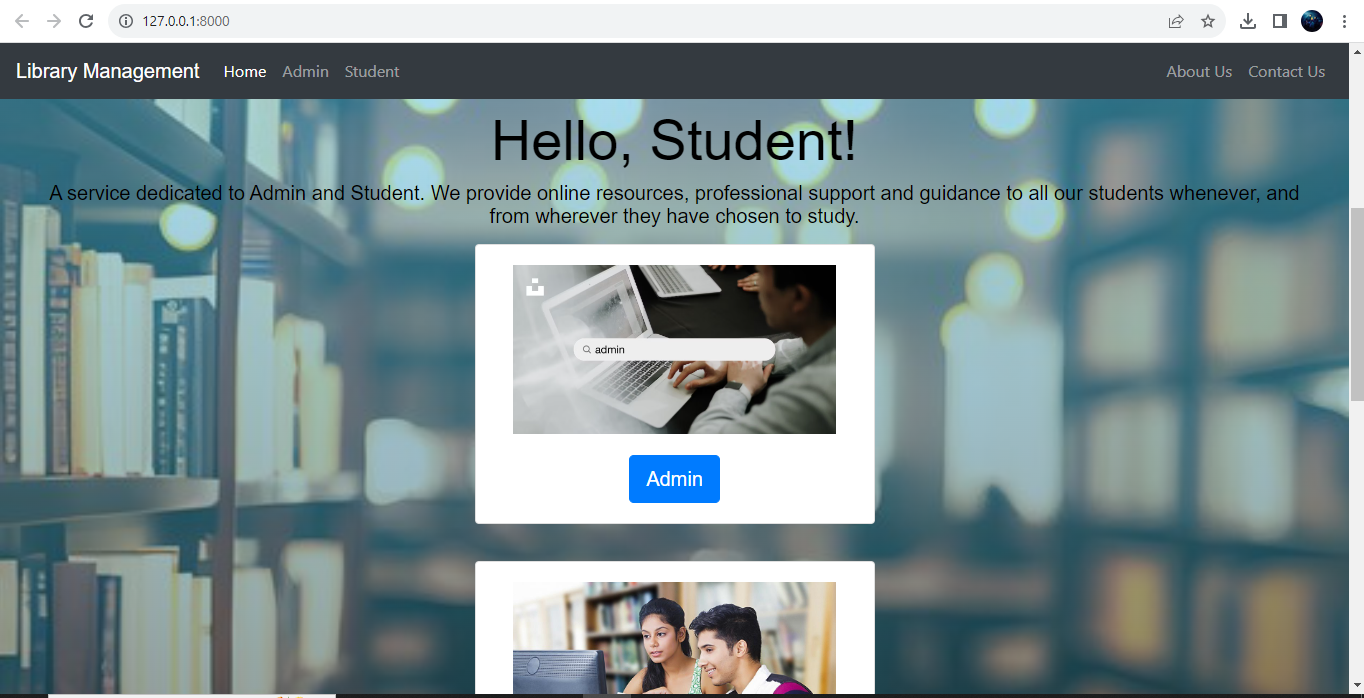
          <img src="{% static "images/pick7.jpg" %}" width="300px" height="200px" border="4">

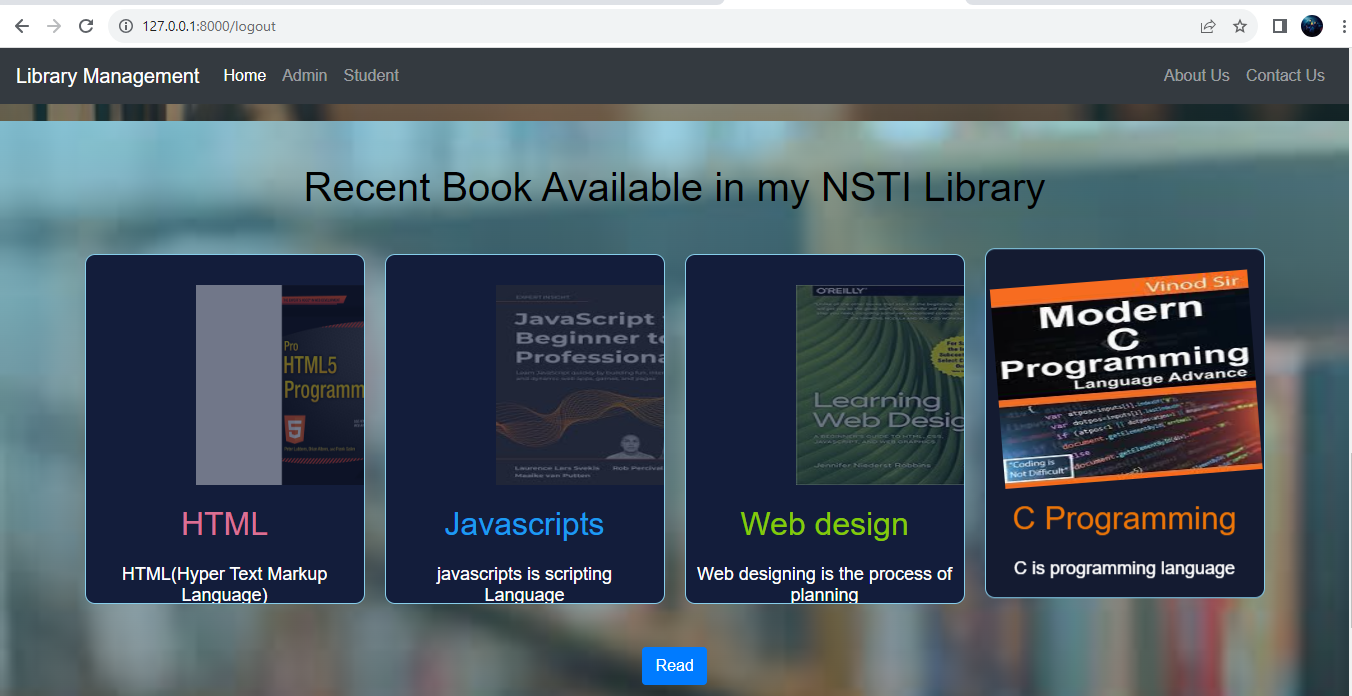
          <img src="{% static "images/pick8.jpg" %}" width="300px" height="200px" border="4">

        </div>   <br><br>  <br><br>

**Here, index output**

****

****

****

**And This my About us page**

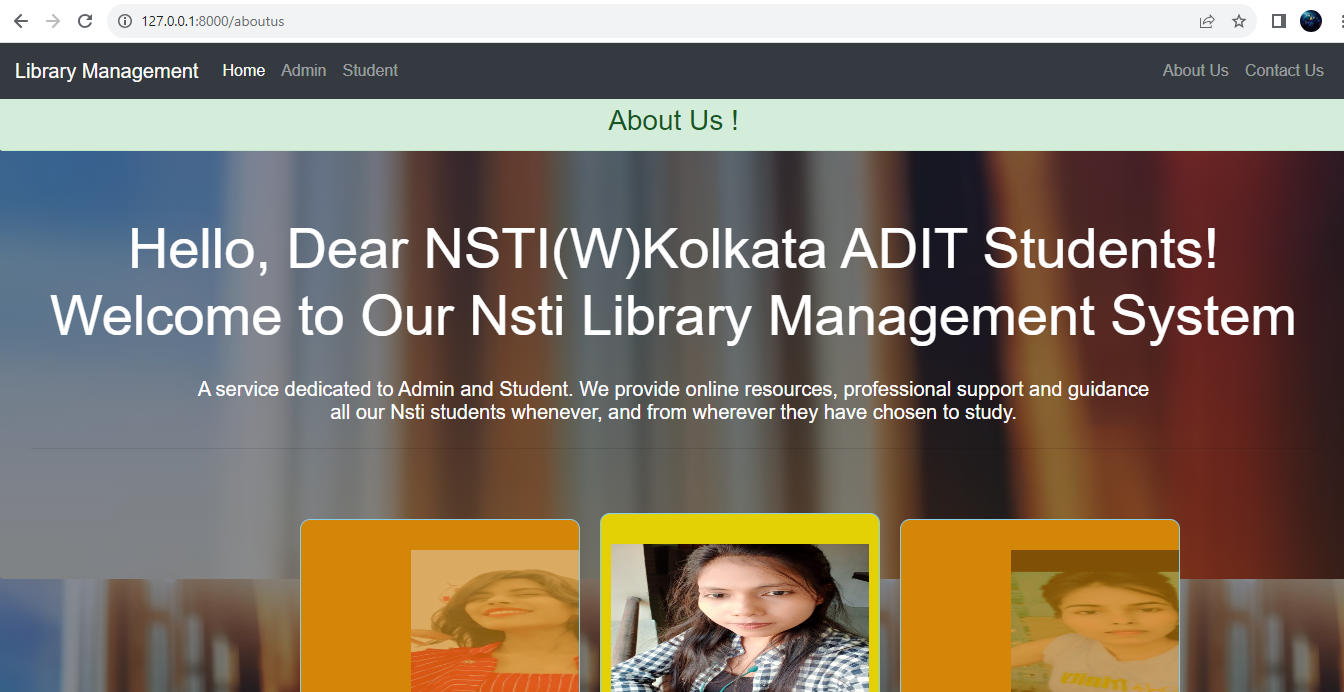
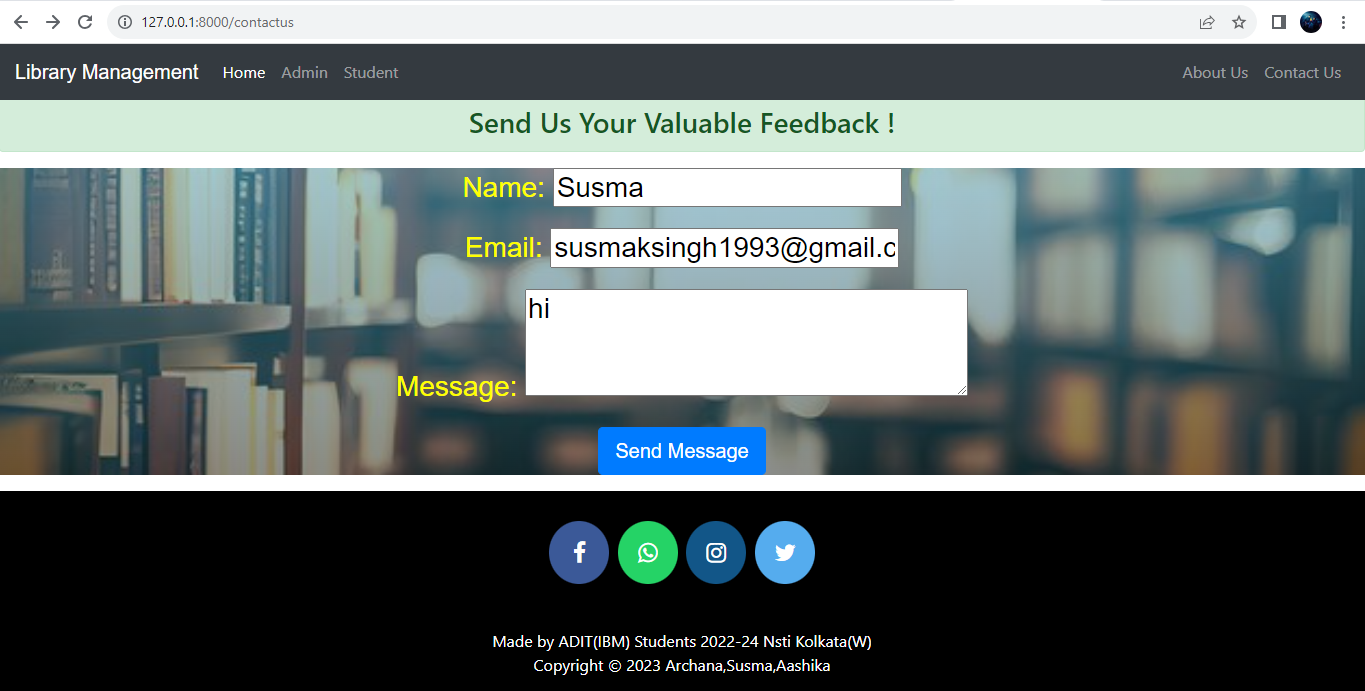
****

Image1:testing if server is running and the page is showing feedback



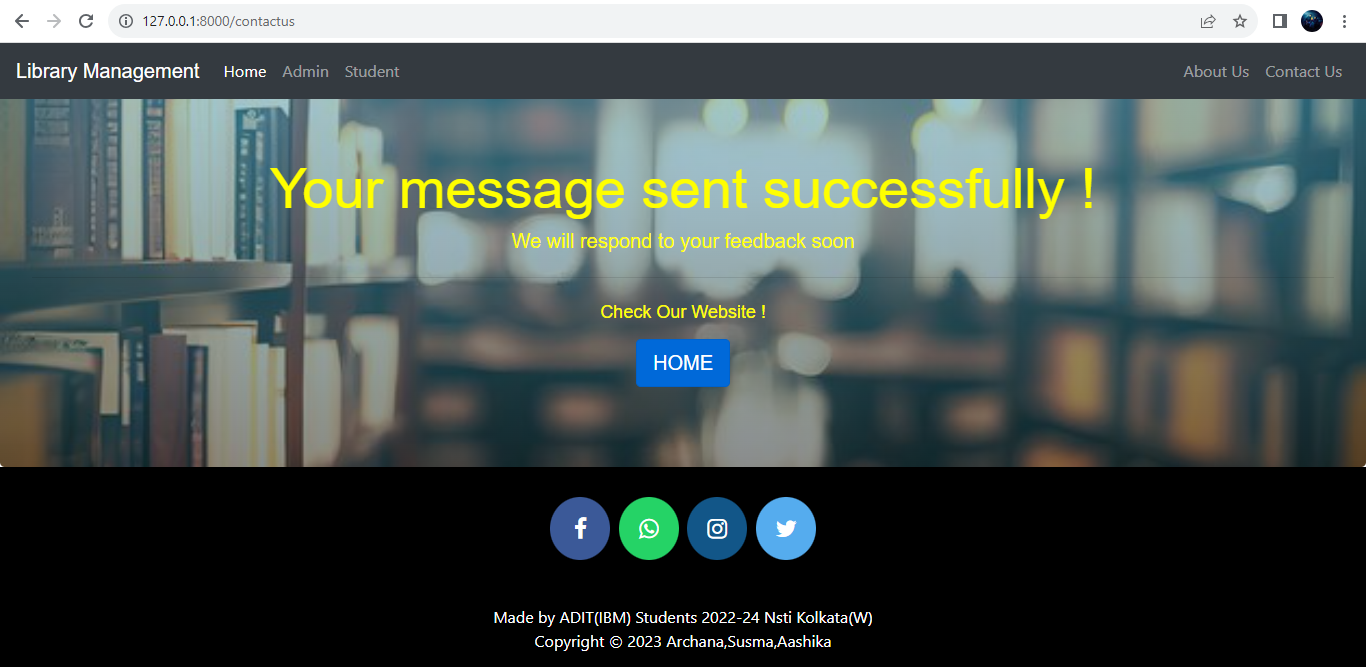
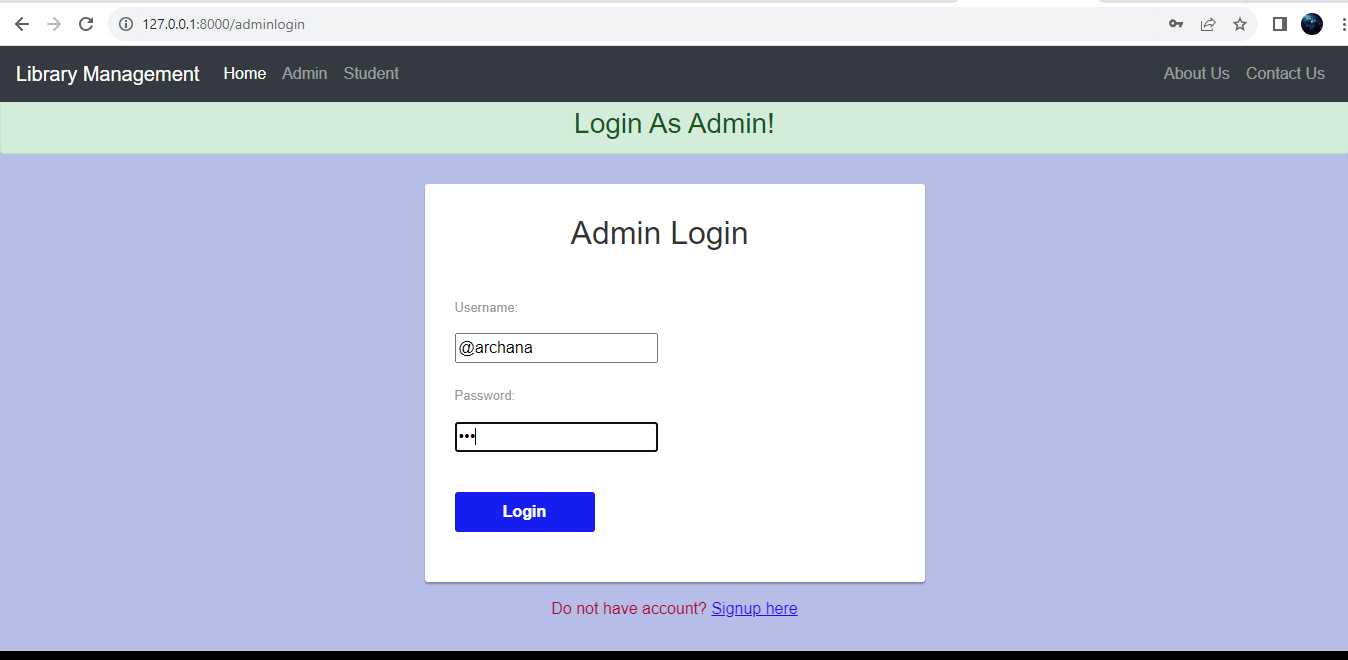
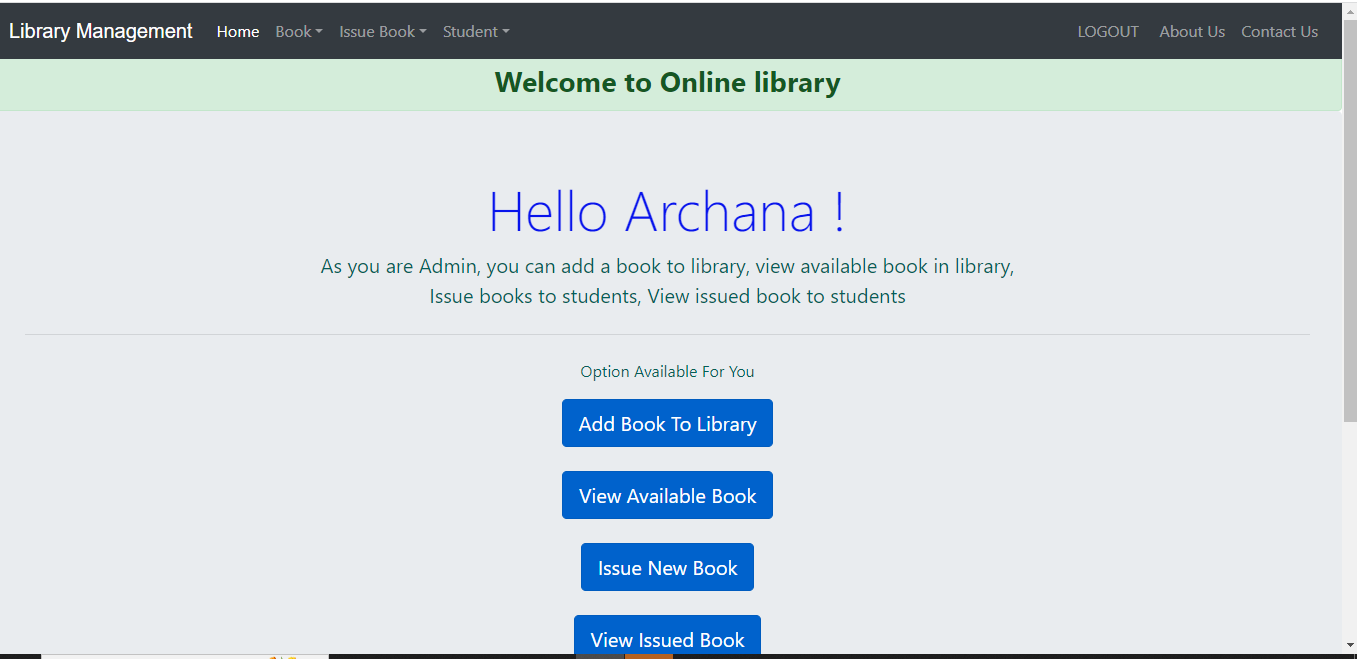
****

Image2: Testing if the feedback forms are working properly

**Admin login page**

****

****

**This is student details output**

**DEPLOYMENT**

Image3: Home page of employee management system

**FUTURE SCOPE**

Here are some of the potential future features that could be added to an library management system project in Django:

* Integration with other systems: The system could be integrated with other systems, such as a human resources system or a payroll system. This would allow for seamless data exchange and make it easier to manage library
* Data analytics: The system could be used to collect and analyse data about students. This data could be used to improve decision-making and identify areas for improvement.
* AI and machine learning: AI and machine learning could be used to improve the system's features and functionality. For example, AI could be used to identify patterns in library data or to automate tasks.

**CONCLUSION**

In conclusion, the development of an library management system project in Django is a significant endeavor with the potential to streamline HR processes, enhance library students, and improve organizational efficiency. This project aims to meet the diverse needs of both HR professionals and students within an organization. Ultimately, the successful implementation of an library management system in Django can lead to increased HR efficiency, better decision-making, improved employee satisfaction, and a more streamlined workforce management process. By aligning the system with the organization's unique requirements and fostering a culture of continuous improvement, it can become a valuable asset in achieving HR and business objectives.

**APPENDIX A**

**SCREENSHOT OF PROJECT**

Image: Home page of library management system

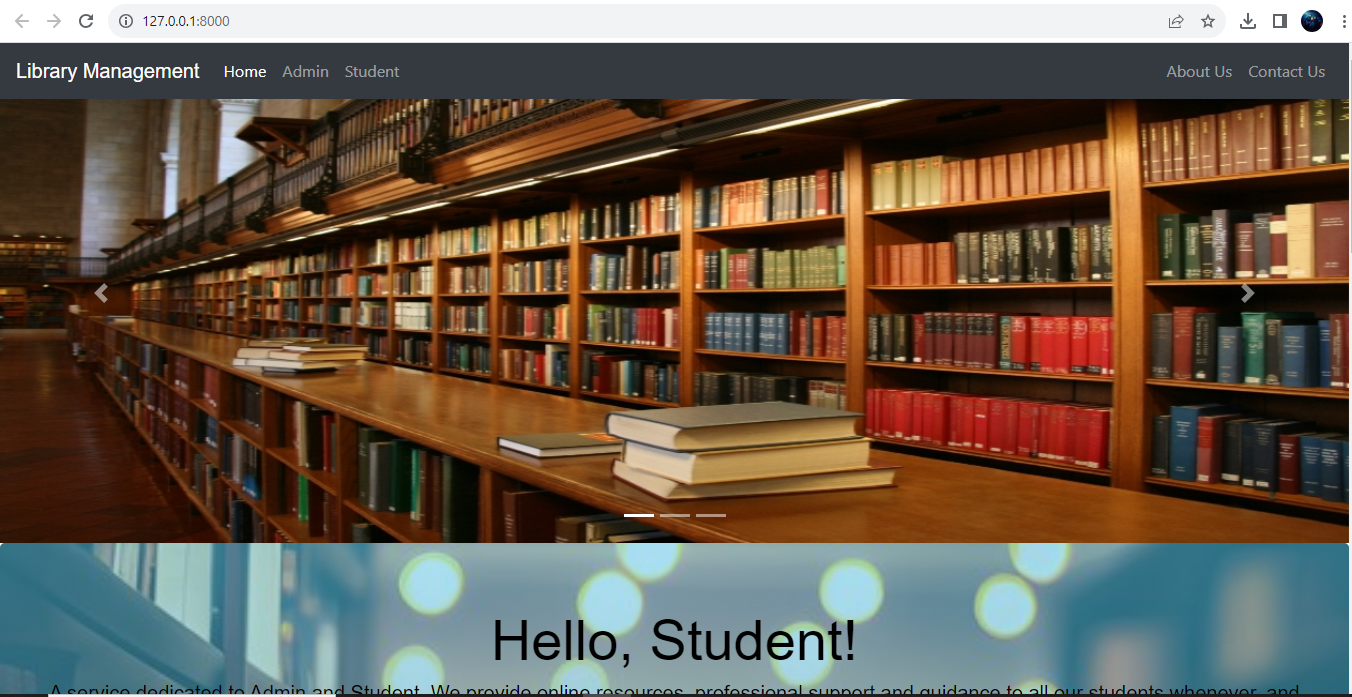
****

Image: Admin page of library management system

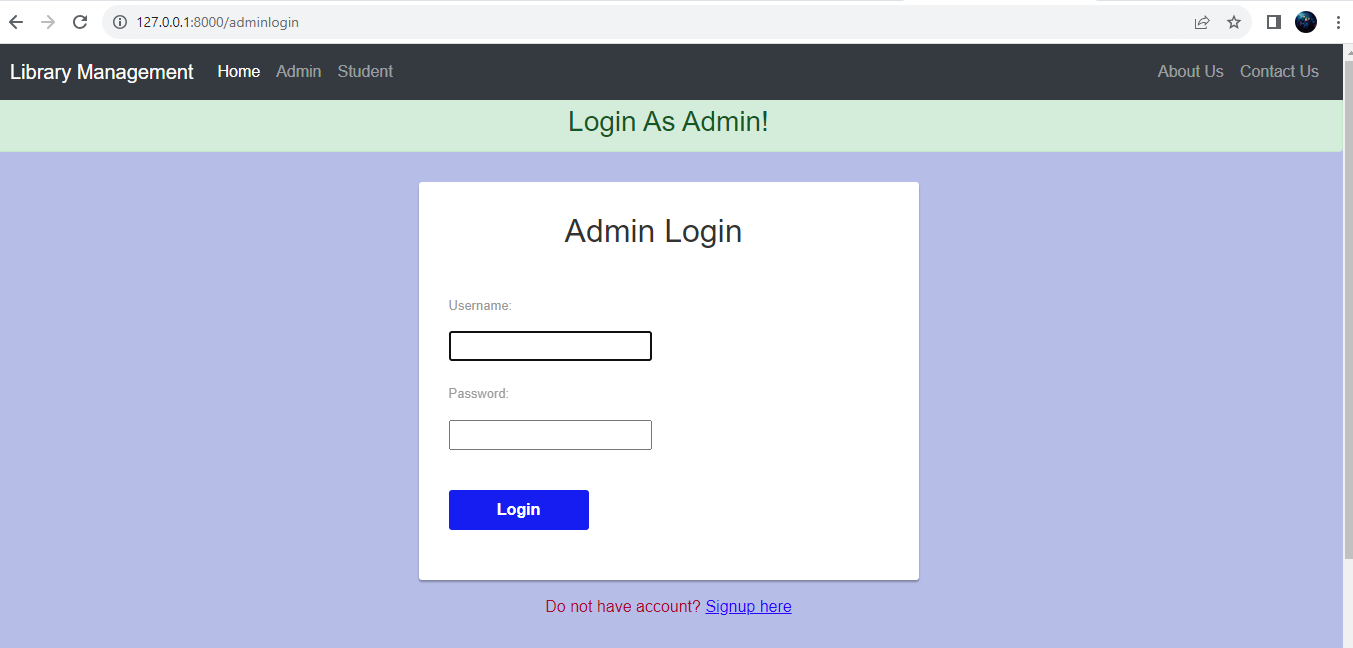


Image: Student login page of library management system

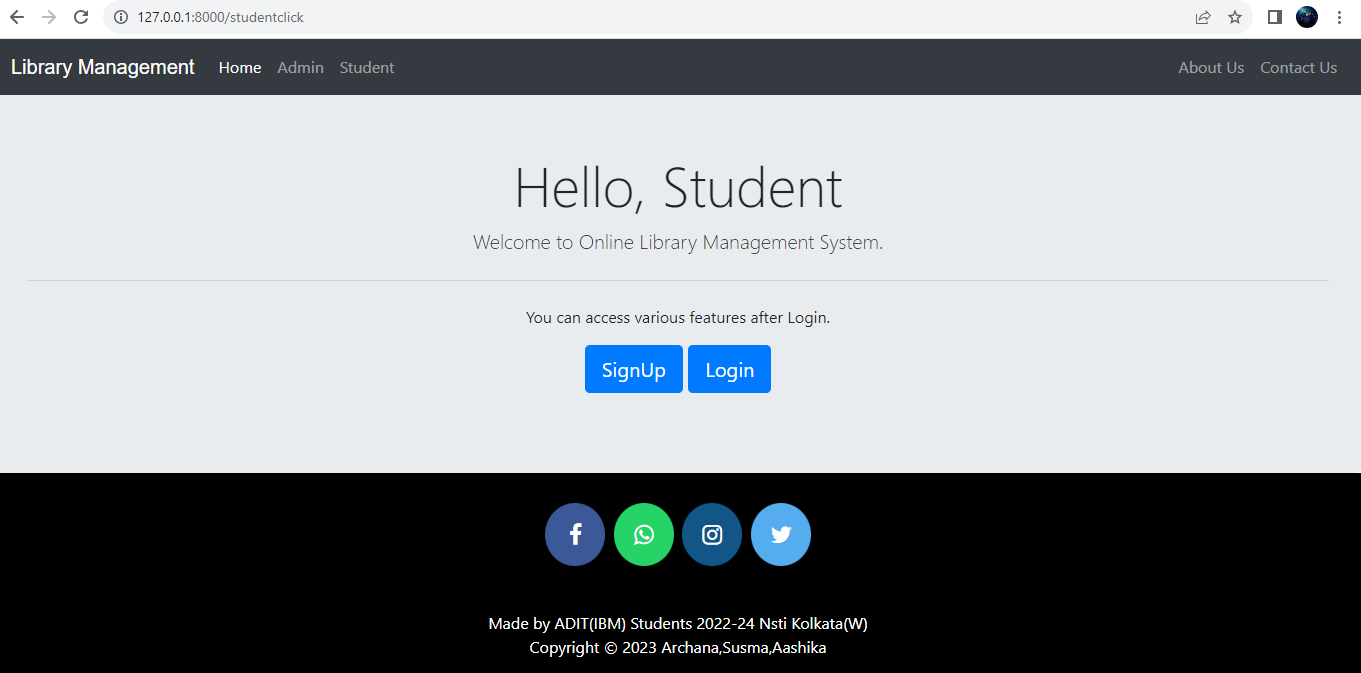


Image: After login Admin page of library management system

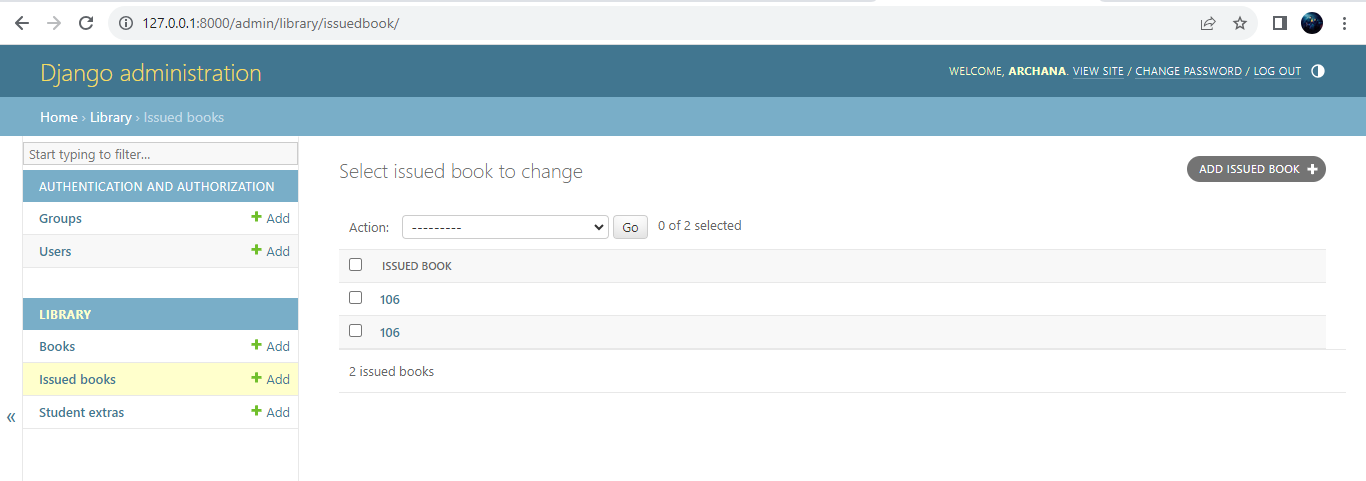
****

Image: Book Details and available library management system

****

Image: student registerd in library management system

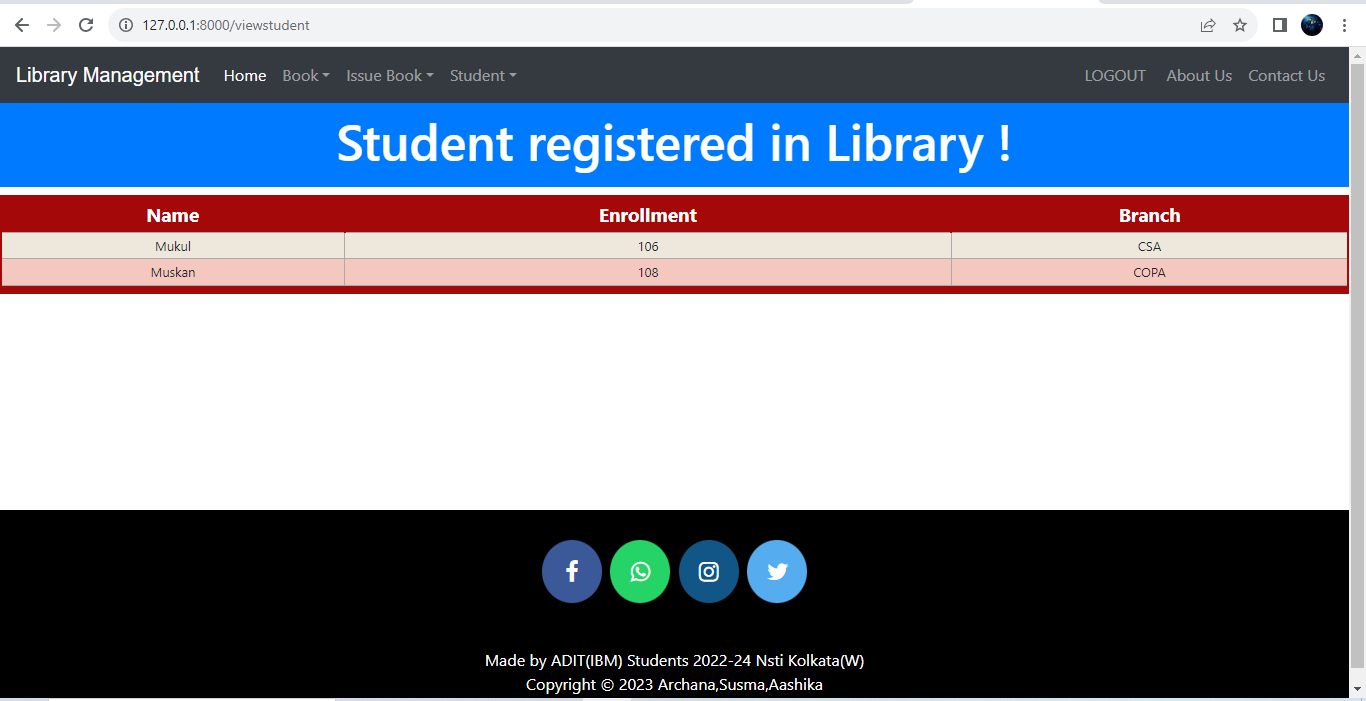
****

Image: issue book to student library management system

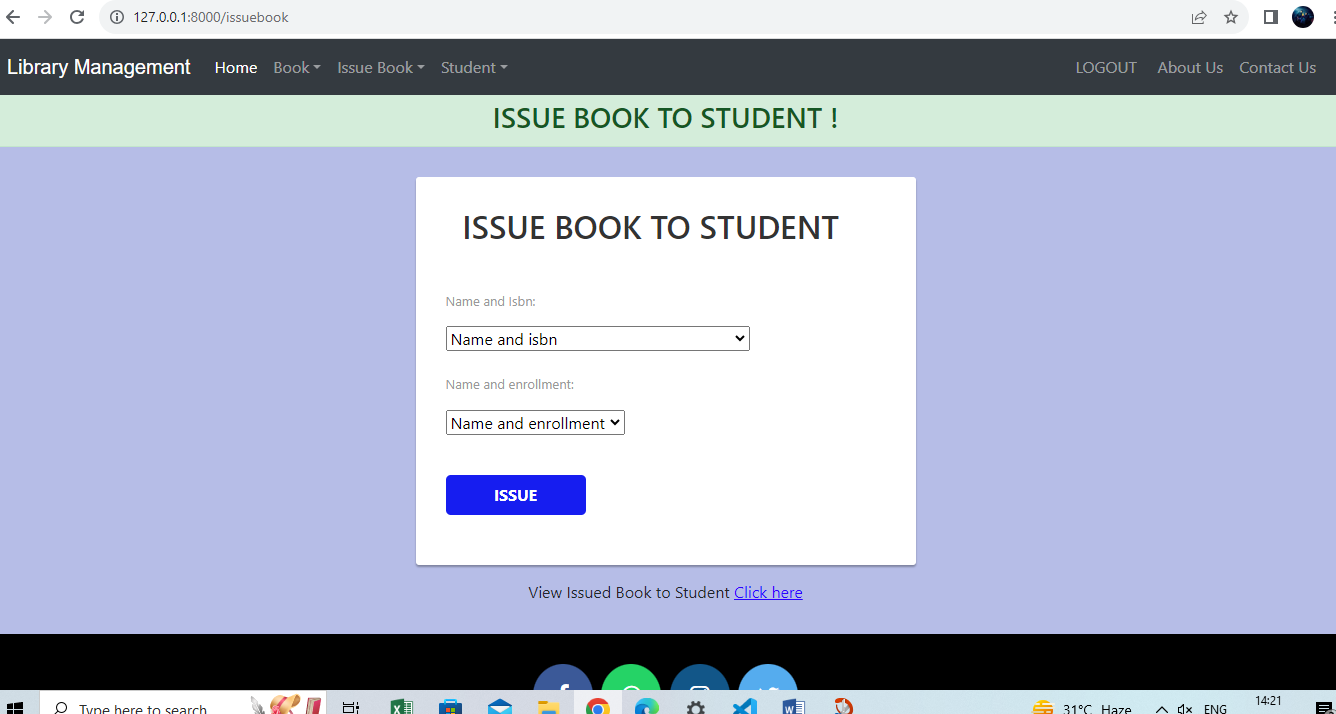
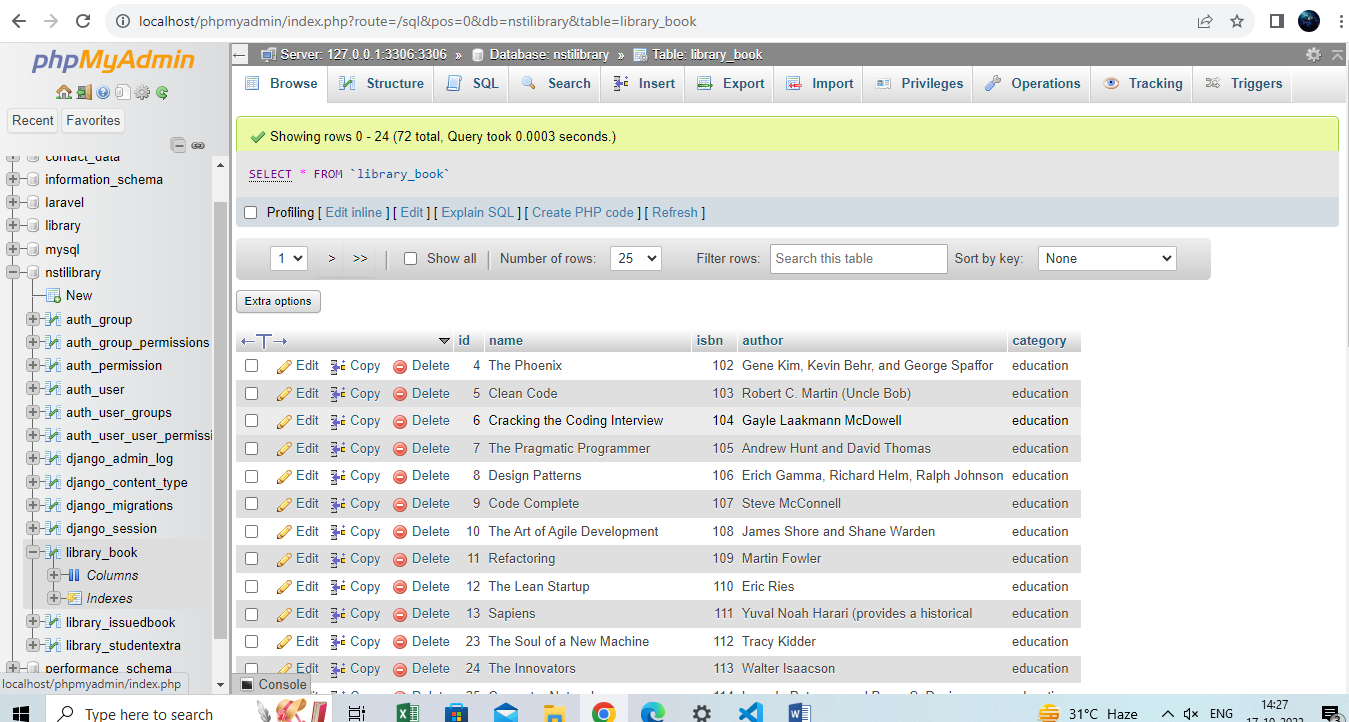
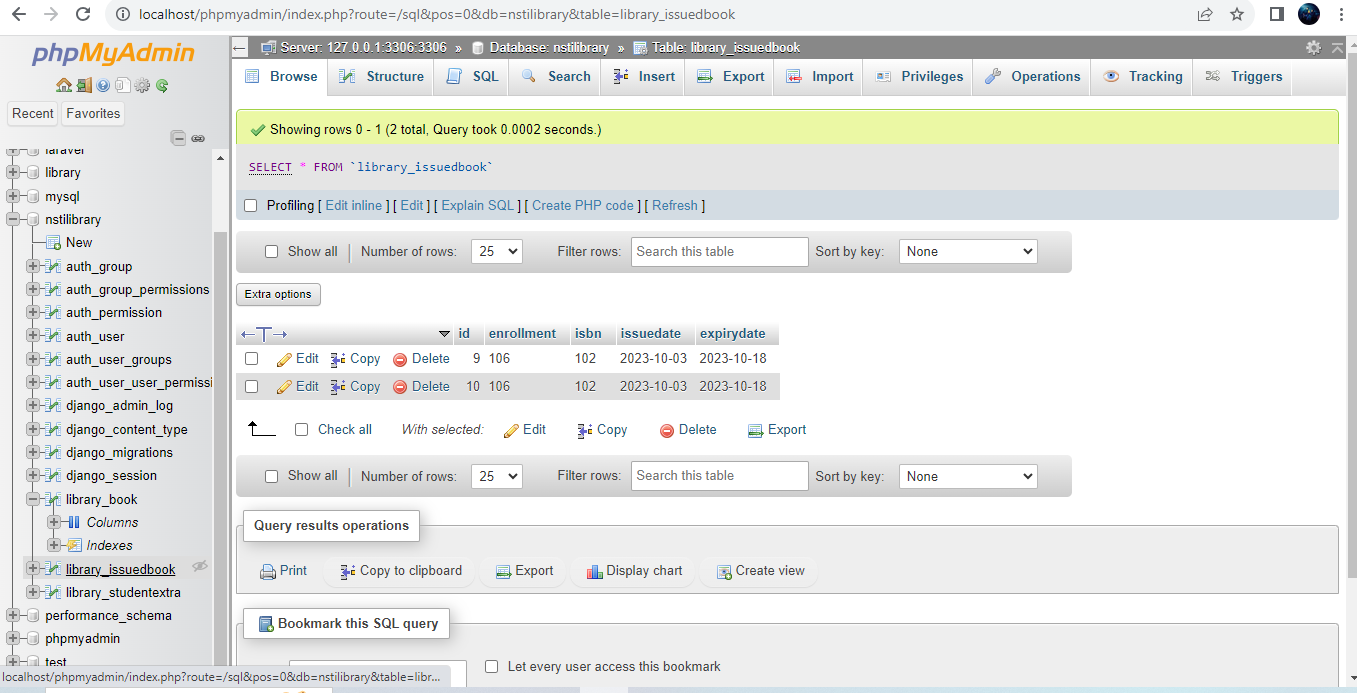
****

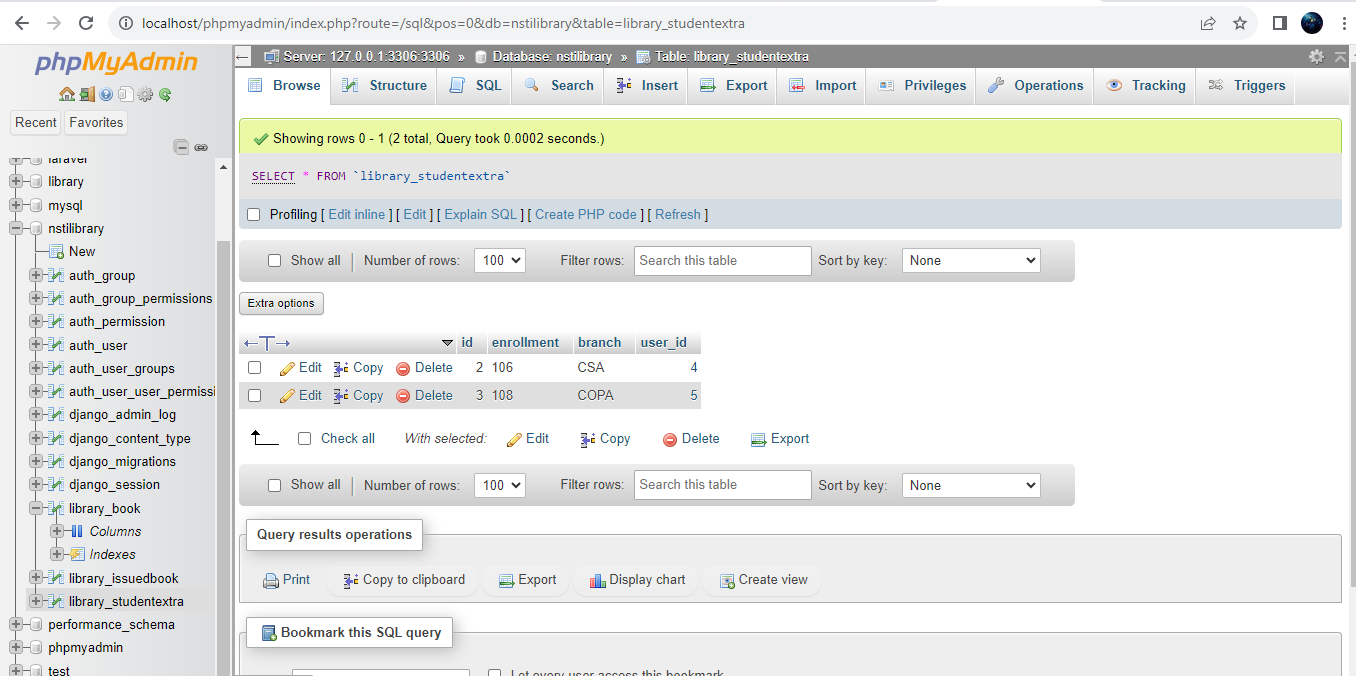
Image10: checking if the employee is removed from page of employee management system

Image: All things are store in database also I provide screenshots

****

**Second screenshot**

****

****

**REFERENCES**

[1] <https://codeshoppy.com/shop/product/employee-django/>

[2] <https://www.youtube.com/watch?v=XMP57hQif4M>